

**WHAT IS CLAIMED IS:**

1. A Java execution device comprising:  
  
an extended class library which includes a class file of a machine code obtained by precompiling a class file included in a standard class library; and  
  
a Java Virtual Machine (JVM) which executes the class file of the machine code class file or an application file included in the extended class library.
2. The Java execution device of claim 1, wherein a machine instruction of the machine code includes an operand in which symbolic reference information is inserted.
3. The Java execution device of claim 2, wherein the Java Virtual Machine (JVM) includes a class linker which converts the symbolic reference information inserted in the operand of the machine instruction into an address.
4. A configuration of a Java class file, wherein the Java class file comprises a constant, a field, and a method, and method information of the method comprises an attribute of a code formed of the machine instruction having the operand in which symbolic reference information is inserted.

5. The configuration of the Java class file of claim 4, wherein the method information further comprises at least one of exception handling information and information used for garbage collection.

6. The configuration of the Java class file of claim 4, wherein the symbolic reference information comprises at least one of information on a constant pool symbol, information on a Java Virtual Machine (JVM)-internal symbol and information on a location of a data block.

7. A method of executing a Java application, the method comprises:

(a) precompiling a class file included in a standard class library into an extended class library file including a machine instruction;

(b) the extended class library file executing the machine instruction; and

(c) executing a Java application file by using at least one of a Just-In-Time (JIT) compiling method and an interpreting method.

8. The method of claim 7, wherein step (a) further comprises inserting symbolic reference information into an operand of the machine instruction.

9. The method of claim 8, wherein step (b) further comprises converting the symbolic reference information inserted in the operand of the machine instruction into an address.

10. A method of precompiling a Java file, the method comprising converting a Java class file or a Java source file into a machine instruction including an operand in which symbolic reference information is inserted.

11. The method of claim 10, wherein the Java class file comprises a standard class file included in a standard Java class library.

12. An execution method in a Java Virtual Machine (JVM), the execution method comprising:

determining whether method information of a method to be executed includes an attribute of a code formed of a machine instruction having a operand in which symbolic reference information is inserted; and

if the method information of the method to be executed includes the attribute of the code formed of the machine instruction, linking the symbolic reference information with an address and executing the machine instruction.

13. The method of claim 12, wherein, if the method information of the method to be executed does not include the attribute of the code formed of the machine instruction, the execution method further comprises one of Just-In-Time (JIT) compiling and interpreting the method.